<u>IN THE CLAIMS:</u>

Please amend the claims as shown immediately below with all changes (e.g., additions, deletions, modifications) included, pursuant to 37 C.F.R. 1.121(c)(1).

Complete listing of the claims:

1-27. (Cancelled).

28.   (Currently Amended) A method of ~~executing a~~ processing transaction routing ~~task~~ tasks, the method including:

receiving a plurality of transaction ~~request~~ requests at an automatic call

distribution system;

generating [[an]] a respective transaction event responsive to receiving each of the

transaction ~~request~~ requests, the transaction event for routing the transaction

request to an agent of the automatic call distribution system;

responsive to the respective transaction ~~event~~ events, identifying a respective

workflow associated with [[the]] each transaction event;

creating a respective task object for each of the transaction events and identified

workflows;

queuing the task objects in a task object queue;

distributing ~~the transaction routing task~~ a task object of the task objects, which at least

partially executes the workflow, from ~~a task~~ the task object queue to an available

thread within a pool of threads operating within a multiprocessor system based

-2-

upon a relative priority of the task objects, ~~the distributing of the transaction~~

~~routing task to the available thread being responsive to dynamically assigning a~~

~~new priority to the transaction routing task~~;

identifying a processor affinity attributed to the distributed task object of the transaction

routing task; and

assigning the available thread to a processor within the multiprocessor system according

to the processor affinity attributed to the transaction routing task to route the

transaction request to the agent of the automatic call distribution system.

29.   (Currently Amended) The method of claim 28, wherein the transaction routing ~~task~~ tasks

includes anyone from a group of transaction routing tasks including receipt of a telephone call,

receipt of a hang up, a request to store data, a request to retrieve data, a request to generate a user

interface for the agent.

30.   (Previously Presented) The method of claim 29, wherein the telephone call includes at

least one of a telephone call received via a public switched telephone network and a voice-over-

IP call received via the Internet.

31.   (Previously Presented) The method of claim 28, wherein the agent includes at least one of

a human agent and a software agent.

32.   (Previously Presented) The method of claim 28, wherein the transaction routing task has a

real-time priority and is distributed in accordance with the real-time priority to the available

thread within the pool of threads.

33.   (Previously Presented) The method of claim 28, including assigning the available thread

to a processor within the multiprocessor system according to a thread priority.

34.   (Previously Presented) The method of claim 33, including assigning the thread priority to

the available thread based on a priority of the transaction routing task distributed to the

available thread.

35.   (Previously Presented) The method of claim 28, further including determining a best

match between the transaction routing task and the available thread.

36.   (Previously Presented) The method of claim 28, wherein the available thread is a

member of a class of threads that are included in the pool of threads, the class of threads being

associated with the priority.

37.   (Currently Amended) A system for ~~executing a~~ processing transaction routing

~~task~~ tasks, the system including:

        an automatic call distribution system to receive a plurality of transaction ~~request~~

requests;

an event subsystem to generate a respective transaction event responsive to receiving each of

the plurality of transaction ~~request~~ requests, the transaction event for routing the

transaction request to an agent of the automatic call distribution system;

a dispatcher to identify a respective workflow associated with each of the transaction

~~event~~ events and that creates a respective task object for each of the plurality of

transaction events and workflows;

a task object queue that contains the respective task objects of the plurality of transaction

requests;

a scheduler ~~to issue the transaction routing task~~ that selects a task object from the task

object queue where the selected task object at least partially executes the workflow

associated with the transaction event, the scheduler to ~~issue~~ select the transaction

routing task from a task queue based upon the relative priority of the selected task

object~~, responsive to a dynamic assignment of a new priority to the transaction~~

~~routing task~~; and

a thread within a pool of threads operating within a multiprocessor system to execute the

selected task object of the transaction routing task, the dispatcher to identify a

processor affinity attributed to the transaction routing task, and to assign the thread

to a processor within the multiprocessor system according to the processor affinity

attributed to the transaction routing task to route the transaction request to the agent

of the automatic call distribution system.

38.   (Previously Presented) The system of claim 37, wherein the dispatcher is to generate the

transaction routing task that at least partially executes the workflow.

39.   (Previously Presented) The system of claim 38, wherein the transaction routing task is dispatched by the dispatcher to the task queue, and wherein the thread within the pool of threads receives the transaction routing task from the task queue.

40.   (Previously Presented) The system of claim 39, wherein the scheduler is to issue the transaction routing task from the task queue to the thread within the pool of threads.

41. (Previously Presented) The system of claim 40, wherein the scheduler is to issue the transaction routing task from the task queue to the thread within the pool of threads based on the priority associated with the transaction routing task.

42.   (Previously Presented) The system of claim 41, wherein the scheduler is to issue the transaction routing task from the task queue according to a real-time priority assigned to the transaction routing task.

43.   (Previously Presented) The system of claim 37, wherein the scheduler is to assign the thread to a processor within the multiprocessor system according to a thread priority.

44.   (Previously Presented) The system of claim 43, wherein the scheduler is to assign the thread priority to the thread based on a priority of the transaction routing task distributed to the

thread.

45. (Previously Presented) The system of claim 37, wherein the scheduler is to determine a

best match between the transaction routing task and the available thread.

46. (Previously Presented) The system of claim 37, wherein the available thread is a member

of a class of threads that are included in the pool of threads, the class of threads being associated

with the priority.

47. (Currently Amended) A system for ~~executing a~~ processing transaction routing ~~task~~ tasks,

the system including:

    a first means to receive a plurality of transaction ~~request~~ requests;

    a second means to generate a respective transaction event responsive to receiving each of

        the transaction ~~request~~ requests, the transaction ~~event~~ events for routing the

        transaction ~~request~~ requests to ~~an agent~~ agents of the first means;

    a third means to identify a workflow associated with each of the transaction ~~event~~ events;

    a task dispatcher that creates a task object for each of the transaction events;

    a task queue that contains the task objects of the plurality of transaction events;

    a fourth means to ~~issue~~ select a task object of the plurity of task objects ~~the transaction~~

        ~~routing task that~~ where the selected task object at least partially executes the

        workflow associated with the transaction event and where selection is based upon a

        relative priority of the plurality of task objects, ~~the fourth means to issue the~~

-7-

transaction routing task from a task queue, responsive to a dynamic assignment of a

new priority to the transaction routing task; and

a fifth means within a pool of threads operating within a multiprocessor system to

execute the ~~transaction routing~~ selected task object, the third means to identify a

processor affinity attributed to the transaction routing task, and to assign the

thread to a processor within the multiprocessor system according to the

processor affinity attributed to the transaction routing task to route the

transaction request to the agent of the first means.

48.   (Currently Amended) A tangible machine readable medium storing a set of instructions

that, when executed by a machine, cause the machine to:

receive a plurality of transaction ~~request~~ requests at a automatic call distribution system;

generate [[an]] a respective transaction event responsive to receiving each of the

plurality of transaction ~~request~~ requests, the transaction ~~event~~ events to route the

transaction ~~request~~ requests to ~~an agent~~ agents of the automatic call distribution

system;

responsive to the transaction ~~event~~ events, identify a respective workflow associated

with each of the plurality of transaction ~~event~~ events;

responsive to the identication of the workflows, creating a task object for each of the

transaction requests;

select and distribute a ~~transaction routing~~ task object of the plurality of task objects,

which at least partially executes the workflow, from a task queue to an available

-8-

thread within a pool of threads operating within a multiprocessor system based upon a relative priority of the task objects, ~~the distribution of the transaction routing task to the available thread responsive to a dynamic assignment of a new priority to the transaction routing task~~;

identify a processor affinity attributed to the selected task object of the transaction routing task; and

assign the available thread to a processor within the multiprocessor system according to the processor affinity attributed to the transaction routing task to route the transaction request to the agent of the automatic call distribution system.

49.   (New) A method of processing a plurality of transaction routing tasks in a multiprocessor transaction processing system having a plurality of processors, such method comprising:

receiving a plurality of transaction requests;

creating a respective task object from each of the transaction events;

queuing the task objects in a task queue;

determining a relative priority among the task objects; and

assigning the task object with a highest relative priority of the plurality of task objects to a first available worker thread within one of the plurality of processors.